International Journal of Natural and Engineering Sciences 4 (3): 27-36, 2010
ISSN: 1307-1149, E-ISSN: 2146-0086, www.nobel.gen.tr

On-line Analysis Out-of-Control Signals for Multivariate Control Chart Using Neural Network

Ardeshir BAHREININEJAD[1] Mohammad Reza Amin NASERI[1] Mojtaba SALEHI[1*] Ali SALMASNIA[1]
[1]Faculty of Engineering, Organization Tarbiat Modares University, Tehran, IRAN

*Corresponding Author
e-mail: m_salehi61@yahoo.com

Received : April 04, 2010
Accepted : June 29, 2010

Abstract

It is common in industrial process to monitor several correlated quality variables simultaneously. Most of multivariate quality control charts are effective in detecting out-of-control signals based upon an overall statistics in multivariate manufacturing processes. The main problem of such charts is that they can detect an out-of-control event but do not directly determine which variable or group of variables has caused the out-of-control signal and what is the magnitude of out of control. This study presents an artificial neural network-based model to supplement the multivariate χ^2 chart. This method consists of two modules. In the first module using a general-neural network, type of unnatural pattern can be recognized. Then by using two special-neural networks for shift mean and trend, it can be recognized magnitude of mean shift and slope of trend for each variable simultaneously. The performance of the proposed approach has been evaluated using a simulated example. The results confirm that the proposed method provides an excellent rate of classification and the output generated by trained network is strongly correlated with the corresponding actual target value for each quality characteristic.

Key word: multivariate manufacturing processes, neural network, χ^2 chart, Statistical process control

INTRODUCTION

Control charts are the most widely applied Statistical process control (SPC) tools used to reveal abnormal variations of monitored measurements. Such charts are useful in identifying the presence of assignable causes in manufacturing processes. A process is out-of-control when points fall outside the control chart limits or the control charts display unnatural (non-random) patterns [1]. Those nonrandom Control Chart Patterns (CCPs) can be associated with a specific set of assignable causes provided that appropriate process knowledge is available. Hence effective identification of nonrandom patterns can greatly narrow down a set of possible assignable causes to be investigated, and significantly speed up the diagnostic process.

Control charts do not provide any pattern-related information when the process is out-of-control. Many supplementary rules, like zone tests or run rules and Expert Systems have also been implemented in Control Chart Pattern recognition (CCPR). But according the reported works, the overall percentages of correctly recognized for these approaches is low. Recently, many studies used Artificial Neural Networks (ANNs) in order to detect patterns more effectively than the conventional approach and their aim is the automatic diagnosis of the patterns. On the other hand, since some advanced automatic data collection and inspection techniques are widely adopted in manufacturing industries. The tasks of SPC traditionally performed by quality engineers have to be automated. Therefore, various artificial intelligence approaches and machine learning techniques have been applied into SPC. Neural Networks (NNs) have excellent noise tolerance in real time, requiring no hypothesis on statistical distribution of monitored measurements. This important feature makes NNs promising and effective tools that can be implemented to improve data analysis in manufacturing quality control applications. In addition, in recognition problems, NNs can recall learned patterns from noisy representations. This feature makes NNs highly appropriate for CCPR because unnatural CCPs are generally contaminated by natural variations in the process. Such applications have been reported to outperform the conventional methods in terms of recognition accuracy and speed.

Pugh [2-3] compared the performance of Shewhart Charts with that of back propagation network (BPN) models in detecting the process mean shift. The trained network generated average run length (ARL) results of around the ARL of an X-bar chart for large shifts, for a sample size of five. Hwarng [4] specified a unique approach for training BPN solely for analyzing cycle patterns on control charts. The recognition output of the network was the amplitude of the cycle pattern, such as 0.5s, 1.0s and 1.5s (where s is the standard deviation of process). Hwarng and Chong [5] proposed an adaptive resonance theory (ART)-based pattern recognizer to detect unnatural CCPs. In their work, the ART-based recognizer could perform fast and cumulative learning due to the unsupervised-learning nature of the ART networks. The ART-based recognizer was superior to the BPN-based recognizer for cycle patterns, inferior for mixture patterns and similar for other patterns. Pham and Chan [6-7] described a type of neural network for control chart pattern recognition. The neural network is self organizing and can learn to recognize new patterns in an on-line incremental manner. Sagiroglu [8] described a type of neural network for speeding up the training process and to compare three training algorithms in terms of speed, performance and parameter complexity for control chart pattern (CCP) recognition. The networks are multilayered perceptrons trained with a resilient propagation, back propagation (BP) and extended delta-bar-delta algorithms. The recognition results of CCPs show the BP algorithm is accurate and provides better and faster results, than other algorithms. Chiu et al. [9] used a BPN to identify shifts in process parameter values according to AR(1) time series models with various autocorrelation coefficients. Their results revealed that NNs successfully separated data that were shifted by one, two and three standard deviations from non-shifted data. The conventional control charts could not identify the same process shifts. Niaki and Abbasi [10] developed a special two levels-based model using T^2 control chart for detecting the out-of-control signals and an MLPNN for identifying the source(s) of the out-of-control signals. Guh, [11] presented a hybrid learning-based model, which integrates NN and DT learning techniques, to detect and discriminate typical unnatural CCPs, while identifying the major parameter (such as the shift displacement or trend slope) and starting point of the CCP detected. Chen et al. [12] presented a hybrid approach by integrating wavelet method with ANNs for on-line recognition of CCPs including concurrent patterns. The main advantage of his approach is its capability of recognizing coexisted or concurrent patterns without training by concurrent patterns. Guh and Shiue [13], proposed a straightforward and effective model to detect the mean shifts in multivariate control charts using decision tree learning techniques. Experimental results using simulation showed that the proposed model couldn't only efficiently detect the mean shifts but also accurately identify the variables that have deviated from their original means.

In many quality control settings, the manufacturing process may have two or more correlated. The usual practice has been to maintain a separate (univariate) chart for each characteristic. Unfortunately, this could result in some fault out-of-control alarms when the characteristics are highly correlated. Hotelling's T^2 statistic might be the most common tool in multivariate analysis for identifying whether the whole process is in out-of-control state. While neural network approaches are generally focused on univariate charts, there are a few multivariate studies found with neural networks in the literature that we review them here briefly. Chen and Wang [14] developed an artificial neural network-based model to supplement the multivariate χ^2 chart. The method not only identifies the characteristic or group of characteristics that cause the signal but also classifies the magnitude of the shifts when the χ^2-statistic signals that mean shifts have occurred. Yu and Xi [15] presented a learning-based model for monitoring and diagnosing out-of-control signals in a bivariate process. In their model, a selective neural network (NN) ensemble approach (DPSOEN, Discrete Particle Swarm Optimization) was developed for performing these tasks. El-Midany et al. [16] proposed a framework for multivariate process control chart recognition. The proposed methodology uses the Artificial Neural Networks (ANNs) to recognize set of subclasses of multivariate abnormal patterns, identify the responsible variable(s) on the occurrence of abnormal pattern and classify the abnormal pattern parameters. In the most presented approaches, recognition problem is limited to identifying the characteristic or group of characteristics that cause the mean shift, some of researches consider the trend also separately, But in this study we try to propose a new approach that can identify shift and trend of quality variables simultaneously and identify the major parameter for each deviated quality variable (magnitude the shift or slope of trend).

To clarify the main problem, Let $X_{ij} = X_{ij1}, X_{ij2}, \ldots, X_{ijp}$ be a p dimension vector that represents the p quality characteristics in the jth observation of the ith subgroup (sample), where $i=1,2,\ldots$ and $j=1,2,\ldots,n$. The lth component of X_{ij}, X_{ij1} denotes the lth quality characteristic, $l=1,2,\ldots,p$. It is assumed that Xij's are independent and identically multivariate normal distribution with known mean μ and covariance matrix $\sum$ when the process is in control. Let $\overline{X_i}$ represent the mean vector for the ith subgroup. The statistic plotted on a multivariate χ^2 control chart for the ith subgroup is given by,

$$\chi_i^2 = n(\overline{X_i} - \mu)\Sigma^{-1}(\overline{X_i} - \mu) \tag{1}$$

When the process is in control, it follows a χ^2 central distribution with p degrees of freedom. Therefore, a multivariate χ^2 control chart can be constructed by plotting χ^2_i versus time with an upper control limit (UCL) given by $\chi^2_{\alpha, p}$ where α is an appropriate significance level for performing the test.

In this research, a new approach for control chart pattern recognition of multivariate observations data is proposed. This method consists of two modules. In the first module at first by a multivariate χ^2 control chart, it can been detected that the process is in control or is out of control then using a general-neural network, type of unnatural pattern (shift or trend) can be recognized. Then by using two special-neural networks for shift mean and trend, it can be recognized magnitude of mean shift or slope of trend for each quality variable simultaneously.

The rest of this research is organized as follows. Section 2 describes proposed method for solving CCPR problem. Section 3 presents a case study CCPR problem that is solved with proposed algorithm and conclusions will be drawn in section 4.

MATERIAL AND METHODS

Since many CCPs often appear separately or together in a manufacturing process, a CCPR system can be developed and trained either as a general-purpose system that can recognize several types of CCP, or as a special-purpose system that can recognize only a particular type of CCP [11]. A suitably developed special-purpose system should be more effective than a general-purpose system for analyzing a particular type of CCP [4]. In this research a modular framework was presented for using advantages of general-purpose and special-purpose simultaneously. Approximately it is impossible that one network would be employed to perform all the required recognition and analysis functions. Because in this situation, the network would have to be complex and convergence in network training would be very difficult. By modular structure we can split the main recognition problem into more manageable sub-problems. In this research module I is a general propose system for recognizing unnatural pattern chart and module II is a special propose system for estimating the major parameters of the unnatural CCPs for all quality variable simultaneously.

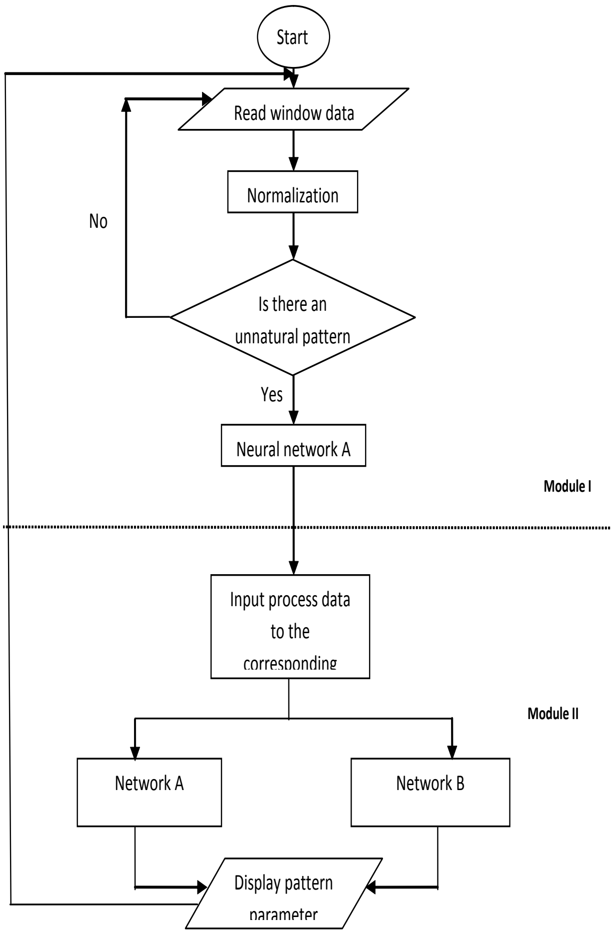


Fig. 1. Proposed approach for CCPR

Module I

In this Module using a multivariate χ^2 control chart and a back propagation neural network various types of unnatural CCPs were detected. If Module I showed a special unnatural pattern, corresponding network in module II was implemented for estimation the major parameters of the unnatural CCPs otherwise collecting of data would be continued.

Data preprocessing

The network input and output range are normally used in Back Propagation Network training to prevent the problem of neuron saturation, which arises from presenting data to a Back Propagation Network in the form of raw data, rather than as data that have been appropriately scaled for the neuro-dynamic functions that are being used. In order to scale down the range of input data for each variable to a new range between -1 and +1, the following equation can be used

$$Normalized\ variate = 2\frac{X - min}{max - min} - 1 \qquad (2)$$

x = input value for a variate,
Min = minimum value in a given set for each variate,
Max = maximum value in a given set for each variate.

Unnatural detection

After data preprocessing the χ^2 value of samples was formed. The χ^2-statistic can reflect the state of the correlation structure and the mean vector of the multivariate process data. In general, control charts have both upper and lower control limits. However, in this case only an upper control limit is used, because extreme values of the χ^2_i statistic correspond to points far remote from the target μ.

The overall quality of a multivariate process can be monitored by comparing the above statistic against a positive UCL ($\chi^2_{\alpha, p}$). These charts are easy to construct, if the process parameters (i.e. μ the mean vector and ∑ the covariance matrix) are already known. If μ is replaced by $\bar{X}$, and ∑ is replaced by S in Equation 1 with n> 1, the $\chi^2_i / c_0\ (p, m, n)$ statistic follows an F-distribution with p and $(mn - m - p + 1)$ degrees of freedom. Here $c_0\ (p, m, n) = [p(m - 1)(n - 1)](mn - m - p + 1)-1$, $\bar{X}$ is the overall sample mean vector and S is the pooled sample variance–covariance matrix. Thus, a multivariate Shewhart control chart for the process mean, with unknown parameters, has the following control limit

$$UCL = \frac{[p(m - 1)(n - 1)]}{(mn - m - p + 1)} F_{1-\alpha, p, (mn-m-p+1)} \qquad (3)$$

This control chart is called a T^2-chart. By using one of these charts we can detect that the process is in control or out of control. However for more detection, we can use neural network A that was explained in the next section.

Unnatural pattern recognition

This network is the responsible for recognizing type of unnatural pattern. It is a general propose neural network that is applied if an unnatural pattern in collected data was detected. For describing the proposed method the stages of application of neural networks including selecting feature vectors, designing the network architecture, training and parameter setting and Performance evaluation are briefly explained below.

Selecting input feature vector

The selection of the feature vector in training set significantly affects the performance of an NN. The input feature vector must be able to intensify the pattern feature of the data set. Most of researches input raw data into a recognizer as an input feature vector but in this study, the statistical features to be extracted from the raw data are added to the input vectors. The use of individual data usually results in a higher Type I error (i.e., shorter in-control ARL), it can reveal an out-of-control situation quickly (i.e., lower Type II error). In-control ARL means the average number of observations that must be taken before an observation indicates an out-of-control condition when the process is actually in control. In a high-speed automatic production scenario, detecting and correcting the inceptive problem as early as possible is important in preventing the possible high-speed manufacturing of defects. Battiti [17] and Smith [18] showed that the BPN integrating raw data and statistical features as input feature vectors has improved performance. Montgomery [1] implemented a two-level resolution IV fractional factorial experimental design for screening and selecting a minimal set of representative statistical features from a list of 10 candidate features. Hassan et al. [19] conducted an experimental study and indicated that the BPN using statistical features as input vectors has better performance than those of the other BPN using raw data as input vectors. In this experiment, we implement Mean, Standard deviation, Skewness, Mean-square value, Autocorrelation and CUSUM with raw data as an input feature vector.

Among various out of control conditions, this study is concerned with mean shifts and trends. For this propose, two neural networks were employed to establish two special-purpose CCP recognizers for mean shifts and trends. When special disturbance at time d_t (zero when no unnatural pattern present) occur at time t_η the observations X of a quality characteristic is expressed as follows:

$$X(t) = \mu + Y(t) + d(t, t_\eta) \quad t \geq t_\eta \tag{4}$$

Where

$X(t)$ quality characteristics measured at time t,

μ process mean vector when the process is in control,

$Y(t)$ $N(0, \sum)$,

$d(t, t_\eta) = u \times b$

u parameter to determine the position of shifting ($u=0$ before shifting, $u=1$ after shifting).

$$b = (k_1\sigma_1, k_2\sigma_2, \ldots, k_p\sigma_p)$$

where k_l is the magnitude of shifts in terms of σ_l, which is the lth quality characteristic. $d(t, t_\eta) = s \times t$

$$s = (r_1\sigma_1, r_2\sigma_2, \ldots, r_p\sigma_p)$$

where r_l is the trend slope in terms of σ_l, which is the lth quality characteristic. Herein each subgroup had ten samples. The feature vector comprises raw observations X_{ij} and the corresponding features.

This study considers seven distinct types of shift and seven distinct types of trend associated with the lth quality characteristic; hence k_l has seven possible values, from -3 to +3 in increments of one and r_l has seven possible values, from -0.3 to +0.3 in increments of 0.1 (shown in Table 3). By having p quality characteristics, $7^p - 1$ types of shift and $7^p - 1$ types of trend can be considered.

In this study, a process with tree variable is considered, which is just an example of a limited case of the general multi SPC when p, the number of variables, equals 3. As an example of in a treevariate process with mean vector (μ_1, μ_2, μ_3) and covariance matrix $\sum$, there are three $(7^3 - 1)$ abnormal classes of patterns for shift and trend. The reference mean vector and covariance matrix for simulated example are:

$$(\mu_1, \mu_2, \mu_3) = (0, 0, 0,), \quad \Sigma = \begin{pmatrix} 1 & 1.5 & 0.5 \\ 1.5 & 3 & 1 \\ 0.5 & 1 & 2 \end{pmatrix}$$

Designing the network architecture

Artificial neural networks consist of numerous interconnected simple processing elements called neurons, which are often organized into a sequence of layers. All layers of the network are linked by weights. These weights are adapted in a supervised or unsupervised learning for exhibiting a desired behavior, by iterating through several input–output vectors. The fundamentals of ANNs can be found in Zurada [20].

The size of the input feature vector that is referred as the identification window size can significantly influence the performance of the proposed model. A small input feature vector will typically detect unnatural patterns more quickly, and may also yield a short in-control ARL (equivalent to a high Type I error). A large window can reduce the recognition efficiency by increasing the time required to detect patterns (higher Type II error, or longer out-of-control ARL). The suitable feature vector size here should balance the Type I and Type II errors. Preliminary experiments were implemented to choose an appropriate size of moving window. A threshold value ($\lambda \in [0,1]$) was implemented to the outputs of two neurons in the output layer. Any value above λ was considered to sign the presence of an unnatural pattern. With an in-control state, a Type I error occurred when any of the output values from the output neurons were equal to or greater than λ. A window size of 12, which corresponds to an in-control ARL of 364 when $\lambda = 0.9$, was selected to give the proposed model a similar in-control ARL to that of a typical Shewhart chart (3 σ limits), which has an in-control ARL of about 350.

This recognizer is a four-layer BPN that has 18p nodes that are used as input data for 12 consecutive points in a control chart (e.g. for each variable, 12 consecutive points + 6 features = 18 nodes). The output layer comprises 2 neurons that they are used for trend and shift. The outputs of the recognizer were scaled within [0, 1], where 1 means that the data are totally fitted to a particular CCP. Two hidden layers both comprise 12p neurons. The experiments showed that the learning results are amended as the number of hidden neurons is increased. Though, increasing the number of hidden neurons over 14 does not amend learning but instead increases the training time. In addition, using more than the necessary number of neurons is also damaging for the network to generalize. The hyper tangent (tansig) function was used as the activation function of the hidden layers and sigmoid (logsig) function was used for output layer.

Training and parameter setting

Effective learning for a Backpropagation Network depends on supplying of enough training examples. In this research, the Monte–Carlo simulation method was used to generate the required data sets of normal and abnormal examples for training and testing. However collecting various data from real-world manufacturing systems is better than Multivariate simulation. Nevertheless, simulation does provide a platform from which investigation into potential problems associated with occurrence and detection of abnormal patterns in a multivariate system can begin [21]. For each CCP parameter setting, 70 examples were generated using CCP example simulator for training. This quantity that can change with number of quality characteristics was determined by an experimental study that revealed that increasing the number of examples did not significantly improve the learning performance. Thus according to types of unnatural patterns 70 (7^p – 1) = 70 (7^3 – 1) = 23940 training examples must be generated.

To ensure the large shifts and trends are detected quickly, the starting point of large shift and trend was set at point 8 to 12 of the recognition window randomly. On the other hand, the starting point of the small shifts and trends was set at the beginning of the recognition window

The initial weights were randomly set between [-0.01, +0.01]. The epochs of the iteration were 350. The learning rate and momentum factor were set to 0.15 and 0.3, respectively. These NN training parameters were set mainly based on trial-and-error experiments performed to determine the best NN training parameters, however for more investigation for parameter setting interested readers are referred to Guh [22] that used the genetic algorithm (GA) to evolve the NN structure, while simultaneously determining a training parameter set (including learning rate, momentum factor, initial range of weights and others) to yield efficiently a near-optimal NN model for the specific application encountered. Barghash and Santarisi [23] also attempted to explore the effect of the training parameters on the performance of the NNs. Their results showed that many parameters usually assigned by experience have significant effect on the performance of the NN.

This study implements the Levenberg–Merquardt Quasi-Network approach because based on Principe et al. [24] research the Levenberg–Merquardt Quasi-Network method is particularly appropriate for training ANNs with Mean Square Error. A convergence condition was established when the classification rate, the number of correctly identified training examples/total number of training examples, exceeded 0.96.

By using MATLAB_ program this proposed network designed and trained. The Network converged after 150 learning epochs with a final CR of 0.96 and an RMS error of 0.0571. The trained network was then tested using 6270 test examples that were also generated by CCP example simulator but with different random seeds. The test result (Table 1) indicated that Network A, exhibited strong capability for recognizing CCPs (CR=0.94).

Table 1. Matrix of test result on the training data for module I

		Required output	
		shift	trend
Network output	shift	0.958	0.067
	trend	0.042	.943

Performance evaluation method

For evaluating the performance of the trained Network *A* in terms of the percentage recognized and ARL 50 example was generated by example simulator for each of unnatural pattern. Table 2 provides Aggregate CR and ARL with different magnitude of shift or trend. For having more practical state in this evaluation by a moving window analysis approach, it is assumed that the process starts under an in-control condition. The initial recognition window contains no unnatural CCP points. Unnatural CCPs begin to appear as the recognition window moves along the time series. The pattern features slowly strengthen as the recognition window moves forwards through the process data stream [11].

If output of this module presented a mean shift we implement neural network *B* for determining magnitude the shift and if presented a trend we implement neural network *C* for determining slop of the trend. If output of this module presented a mean shift and trend simultaneously, we implement both neural network *B* and *C*.

Table 2. Evaluation results of neural network A

shift	Aggregate CR with different magnitude of shift or trend	Aggregate ARL with different magnitude of shift or trend
1nd quality characteristic	0.946	7.35
2nd quality characteristic	0.937	7.82
3nd quality characteristic	0.951	7.63
1nd and 2nd quality characteristic	0.958	5.54
1nd and 3nd quality characteristic	0.966	5.81
2nd and 3nd quality characteristic	0.956	5.63
1nd and 2nd and 3nd quality characteristic	0.974	4.62
trend		
1nd quality characteristic	0.894	9.05
2nd quality characteristic	0.873	8.86
3nd quality characteristic	0.901	9.43
1nd and 2nd quality characteristic	0.948	6.50
1nd and 3nd quality characteristic	0.926	7.31
2nd and 3nd quality characteristic	0.961	6.73
1nd and 2nd and 3nd quality characteristic	0.9684	6.12

Module II

This Module is the responsible on identification the key parameter of pattern (e.g. shift magnitude and slop trend). It consists of two separated specialist-NNs for identification the key pattern parameter. The used specialist-NN will be selected via according to output of the previous module. Any trained network can recognize deviated variables.

In this approach three-layer fully connected feed-forward networks are used which implement the back-propagation training rule developed by Rumelhart, McClelland, and PDP Research Group [25]. Some adjustments were made to improve the performance of the network in solving the problem considered herein.

Designing the network architecture

In this study a three-layer fully connected feed-forward network with aback-propagation training algorithm was implemented. Fig. 2 shows the neural network architecture, which includes an input layer with $16p$ nodes that are used as input data for 10 consecutive points in a control chart, a hidden layer with $6p$ nodes and an output layer with p nodes. Each output node represents a unique target value associated with a type of shift or trend for a particular quality characteristic. Table 3 shows the relationship between target value and type of shift and trend associated with a particular quality characteristic. Cheng [26] claimed that the hyper tangent transfer function effectively detects process changes in different directions. The hyper tangent (tansig) function was used as the activation function of the hidden and output layers of NNs in Module II.

A large recognition window corresponds to higher recognition accuracy. The minimal window size that yielded satisfactory identification accuracy was determined empirically to be 30 (10 nodes for three variables or $10p$). Many theoretical and simulative investigations of engineering applications have demonstrated that the number of hidden layers need not exceed two [27]. Since one hidden layer can approximate any continuous mapping from the input patterns to the output patterns in backpropagation network, one hidden layer was considered. The optimal number of node depends on the problem. If a network has too few hidden neurons, it cannot learn the training set and cannot generalize well. On the other hand, a network with too many neurons can tend to memorize the training set and will also damage the ability of the network to generalize. In this study, the number of hidden neurons was selected based on trial and error experiments. The proposed neural network was showed in Fig. 2.

Table 3. Relationship between target value and type of shift for the lth quality characteristic

Target value	Type of shift for the lth quality characteristic	Type of trend for the lth quality characteristic
+1	$+ 3\sigma$	$+ 0.3\sigma$
+0.7	$+ 2\sigma$	$+ 0.2\sigma$
+0.4	$+ 1\sigma$	$+ 0.1\sigma$
0	0	0
-0.4	$- 1\sigma$	$- 0.1\sigma$
-0.7	$- 2\sigma$	$- 0.2\sigma$
-1	$- 3\sigma$	$- 0.3\sigma$

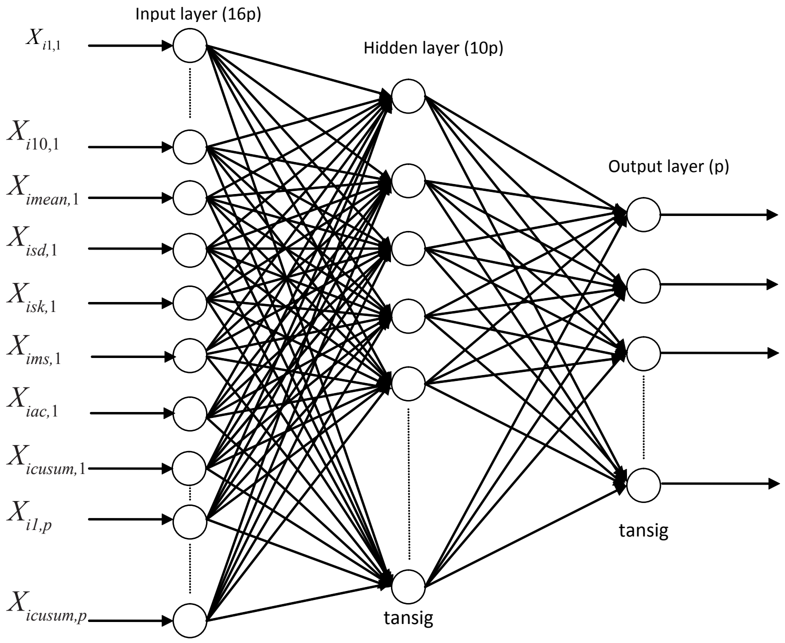


Fig. 2. shows the neural network architecture

Training and parameter setting

To train the two special neural networks, 50 examples were generated using CCP example simulator for each type of unnatural pattern. This quantity that can change was determined by an experimental study that revealed that increasing the number of examples did not significantly improve the learning performance. Thus according to types of unnatural patterns 50 $(7^p - 1)$ = 50 $(7^3 - 1)$ = 17100 training examples must be generated.

The initial network connection weight, learning rate and momentum coefficient were set as in Module I. During training, the convergence condition was reached within 100 training epochs with a confusion matrix coefficient 0.95. The BPN were trained by the implementation of back-propagation algorithm in MATLAB toolbox.

This module implements the Levenberg–Merquardt Quasi-Network approach as training algorithm. The input vectors are implemented to the network and propagated forward to yield the output. Connection weights were adjusted after each pattern was presented. The mean square error (MSE) associated with the output layer is propagated backward through the network, by modifying the weights. Since numbers of example is too many Table 4 and Table 5 details only some of the test examples.

Table 4. Matrix of shift test result on the training data for module II

shift	Type of unnatural pattern	Target value	Average output of major parameter special network		
			mean	Aggregate Standard deviation	errors
1nd quality characteristic	(+2,0,0)	(+0.7,0,0)	(+0.73,+0.08,+0.11)	0.180	(0.03,0.08,0.11)
	(-2,0,0)	(-0.7,0,0)	(-0.64,-0.03,+0.09)	0.203	(0.04,0.03,0.09)
2nd quality characteristic	(0,+2,0)	(0,+0.7,0)	(+0.03,+0.74,+0.06)	0.196	(0.03,0.04,0.06)
	(0,-2,0)	(0,-0.7,0)	(-0.07,-0.75,+0.01)	0.222	(0.07,0.05,0.01)
3nd quality characteristic	(0,0,+2)	(0,0,+0.7)	(-0.09,-0.07,+0.67)	0.214	(0.09,0.07,0.03)
	(0,0,-2)	(0,0,-0.7)	(+0.13,+0.09,-0.78)	0.219	(0.13,0.09,0.08)
1nd and 2nd quality characteristic	(+2,-2,0)	(+0.7,-0.7,0)	(+0.64,-0.69,+0.03)	0.195	(0.06,0.01,0.03)
	(-2,+2,0)	(-0.7,+0.7,0)	(-0.75,+0.71,-0.08)	0.224	(0.05,0.01,0.08)
1nd and 3nd quality characteristic	(+2,0,-2)	(+0.7,0,-0.7)	(+0.69,-0.10,-0.74)	0.234	(0.01,0.1,0.04)
	(-2,0,+2)	(-0.7,0,+0.7)	(-0.74,+0.06,+0.72)	0.246	(0.04,0.06,0.02)
2nd and 3nd quality characteristic	(0,+2,-2)	(0,+0.7,-0.7)	(+0.07,+0.67,-0.64)	0.216	(0.07,0.03,0.06)
	(0,-2,+2)	(0,-0.7,+0.7)	(+0.08,-0.73,+0.67)	0.240	(0.08,0.03,0.03)
1nd and 2nd and 3nd quality characteristic	(+2,-2,+2)	(+0.7,-0.7,+0.7)	(+0.74,-0.75,+0.64)	0.289	(0.04,0.05,0.06)
	(-2,+2,-2)	(-0.7,+0.7,-0.7)	(-0.69,+0.76,-0.77)	0.273	(0.01,0.06,0.07)

Table 5. Matrix of trend test result on the training data for module II

Trend	Type of unnatural pattern	Target value	Average output of major parameter special network		
			mean	Aggregate Standard deviation	errors
1nd quality characteristic	(+0.2,0,0)	(+0.7,0,0)	(+0.67,-0.07,+0.09)	0.169	(0.03,0.07,0.09)
	(-0.2,0,0)	(-0.7,0,0)	(-0.74,-0.13,+0.12)	0.198	(0.04,0.13,0.12)
2nd quality characteristic	(0,+0.2,0)	(0,+0.7,0)	(+0.10,+0.74,-0.06)	0.231	(0.1,0.04,0.06)
	(0,-0.2,0)	(0,-0.7,0)	(+0.08,-0.74,+0.08)	0.206	(0.08,0.04,0.08)
3nd quality characteristic	(0,0,+0.2)	(0,0,+0.7)	(-0.07,+0.09,+0.77)	0.257	(0.07,0.09,0.07)
	(0,0,-0.2)	(0,0,-0.7)	(-0.07,-0.11,-0.67)	0.243	(0.07,0.11,0.03)
1nd and 2nd quality characteristic	(+0.2,-0.2,0)	(+0.7,-0.7,0)	(+0.66,-0.78,+0.03)	0.285	(0.04,0.08,0.03)
	(-0.2,+0.2,0)	(-0.7,+0.7,0)	(-0.75,+0.71,+0.11)	0.176	(0.05,0.01,0.11)
1nd and 3nd quality characteristic	(+0.2,0,-0.2)	(+0.7,0,-0.7)	(+0.76,-0.09,-0.67)	0.213	(0.06,0.09,0.03)
	(-0.2,0,+0.2)	(-0.7,0,+0.7)	(-0.73,-0.15,+0.79)	0.246	(0.03,0.15,0.09)
2nd and 3nd quality characteristic	(0,+0.2,-0.2)	(0,+0.7,-0.7)	(-0.12,+0.73,-0.73)	0.261	(0.12,0.03,0.03)
	(0,-0.2,+0.2)	(0,-0.7,+0.7)	(+0.05,-0.79,+0.71)	0.232	(0.05,0.09,0.01)
1nd and 2nd and 3nd quality characteristic	(+0.2,-0.2,+0.2)	(+0.7,-0.7,+0.7)	(+0.64,-0.79,+0.68)	0.275	(0.06,0.09,0.02)
	(-0.2,+0.2,-0.2)	(-0.7,+0.7,-0.7)	(-0.76,+0.65,-0.76)	0.291	(0.06,0.05,0.06)

Performance evaluation of Module II

For evaluating the performance of module II the simulated examples were run. This performance evaluation was performed at the same time as that of Module I, which assumes that the process begins in an in-control condition, was applied in the evaluation process herein.

Table 6. Evaluation results of neural network ***B*** & ***C***

	average error	Aggregate Standard deviation of average error
shift	(0.0731, 0.0728, 0.0723)	0.0216
trend	(0.0741, 0.0758, 0.0731)	0.0231

Table 6 summarizes the average errors and Aggregate Standard deviation of average error of the CCP parameter identification. Table 6 shows that the overall performance of CCP parameter identification is reasonably good. Notably, the results here were calculated only for the CCPs whose type was correctly recognized by Module I.

Case study

For implementing the proposed approach, consider the part that is shown in Fig. 3. In this part the plans A and B shown are finished on the same machine having the same tool holder. This part is assembled with other parts and these two heights are correlated to each other with the correlation coefficient 0.65. These two heights are considered as χ_1 and χ_2, thus we have a control chart pattern recognition problem with two variables.

Fig. 4 shows the heights of 35 consecutive batches of the part for each quality characteristic respectively. The in-control mean and standard deviation of the process for plan A are 0.3750, 0.0062 and for plan B are 0.8130, 0.0096 respectively. The UCL and LCL are calculated to be 0.3936 and 0.3564 for χ_1, 0.8418 and 0.7842 for χ_2 respectively. An assignable cause a shift the mean χ_1 to 0.3874 (displacement of the mean = $+2\sigma$) at sample No. 25 and a trend for χ_2 (slope of trend = -2σ) at sample No. 25 too. Fig. 4 shows the individual control charts and Fig. 5 shows χ^2 control chart for these observations.

If we presented collected data to the proposed model in real-time with window size 10, Output of network A shown in Table 7 detected an unnatural pattern in 33-th observation. We can see a shift in χ_1 and trend in χ_2. With these simultaneous deviations (trend and shift) in quality characteristics, ones may be able to recognize cause of deviations but for more accurate results outputs of network B and C must be analyzed. Network B recognizes a shift magnitude $+2\sigma$ approximately in χ_1 and network C recognizes a trend slope -0.2σ approximately in χ_2. These results are highly correlated with real situations.

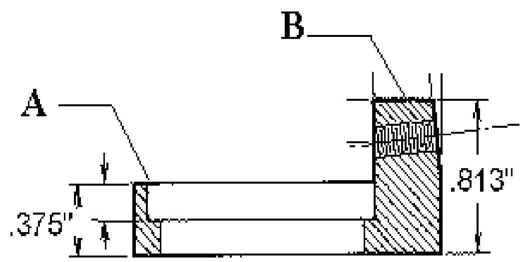
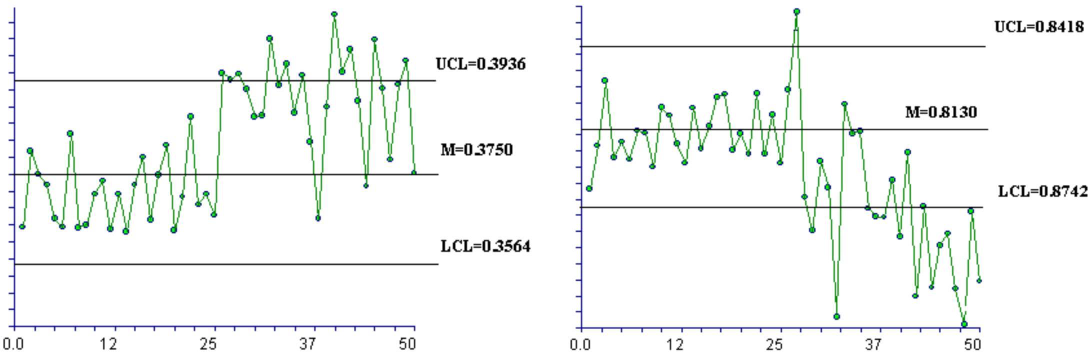


Fig. 3. Plans A and B in typical part.



a) Displacement of the mean is $+2\sigma$ for χ_1 b) slope of trend is -2σ for χ_2

Fig. 4. Individual control chart for χ_1 and χ_2 for presented part

Table 7. outputs of network A, B and C for case study

NO.	Output A	Output B	Output C	NO.	Output A	Output B	Output C
1-10	(0.0896, 0.0225)	-	-	21-30	(0.5042, 0.5333)	(+0.43, -0.04)	(+0.08,0.53)
2-11	(0.0067, 0.0659)	-	-	22-31	(0.6176, 0.6674)	(+0.52, +0.06)	(+0.18, 0.49)
3-12	(0.0665, 0.0561)	-	-	23-32	(0.8015, 0.7113)	(+0.51, +0.10)	(-0.03, 0.51)
4-13	(0.0099, 0.0084)	-	-	24-33	(0.9118, 0.8155)	(+0.56, -0.13)	(+0.07, 0.55)
5-14	(0.0428, 0.0329)	-	-	25-34	(0.9191, 0.9185)	(+0.62, -0.07)	(-0.05, 0.63)
6-15	(0.0190, 0.0246)	-	-	26-35	(0.9259, 0.9144)	(+0.78, +0.06)	(+0.03, 0.62)
7-16	(0.0190, 0.0246)	-	-	27-36	(0.9239, 0.9278)	(+0.70,+0.09)	(+0.02, 0.68)
8-17	(0.0428, 0.0329)	-	-	28-37	(0.9275, 0.9346)	(+0.69,-0.01)	(-0.08, 0.71)
9-18	(0.0212, 0.0283)	-	-	29-38	(0. 9213, 0.9379)	(+0.71,-0.02)	(-0.10, 0.72)
10-19	(0.0216, 0.0264)	-	-	30-39	(0.9279, 0.9434)	(+0.77,+0.12)	(-0.11, 0.70)
11-20	(0.0431, 0.0245)	-	-	31-40	(0.9375, 0.9448)	(+0.63,+0.03)	(-0.05, 0.71)
12-21	(0.0190, 0.0246)	-	-	32-41	(0.9465, 0.9425)	(+0.67,-0.06)	(-0.06, 0.72)
13-22	(0.0833, 0.0258)	-	-	33-42	(0.9412, 0.9614)	(+0.66,+0.03)	(+0.01, 0.75)
14-23	(0.0330, 0.0131)	-	-	34-43	(0.9636, 0.9673)	(+0.64,-0.04)	(+0.05, 0.71)
15-24	(0.0951 0.0282)	-	-	35-44	(0.9675, 0.9546)	(+0.71,-0.01)	(-0.09, 0.68)
16-25	(0.0431, 0.0245)	-	-	36-45	(0.9532, 0.9562)	(+0.72,+0.08)	(+0.05, 0.73)
17-26	(0.1274, 0.1676)	-	-	37-46	(0.9634, 0.9564)	(+0.70,+0.09)	(-0.03, 0.68)
18-27	(0.2282, 0.1773)	-	-	38-47	(0.9712, 0.9614)	(+0.71,-0.01)	(+0.06, 0.69)
19-28	(0.3246, 0.3205)	-	-	39-48	(0.9636, 0.9785)	(+0.68,+0.04)	(-0.02, 0.72)
20-29	(0.4192 0.4130)			40-49	(0.9675, 0.9805)	(+0.73,+0.03)	(-0.09, 0.75)

CONCLUSION

Unnatural CCPs provide clues to potential quality problems at an early stage, to eliminate defects before they are produced. A modular based neural network was presented for control chart pattern recognition problem herein that can be used for multi variate process. general-neural network was implemented in the first module for recognizing, type of unnatural pattern. Then two special-neural networks for shift mean and trend were implemented for recognizing magnitude of mean shift and slope of trend for each variable simultaneously. The performance of the proposed approach has been evaluated using a simulated example and case study. The results confirm that the proposed method provides an excellent rate of classification and the output generated by trained network is strongly correlated with the corresponding actual target value for each quality characteristic. The main contribution of this study is that model can identify shift and trend of quality variables simultaneously and identify the major parameter for each deviated quality variable (magnitude the shift or slope of trend).

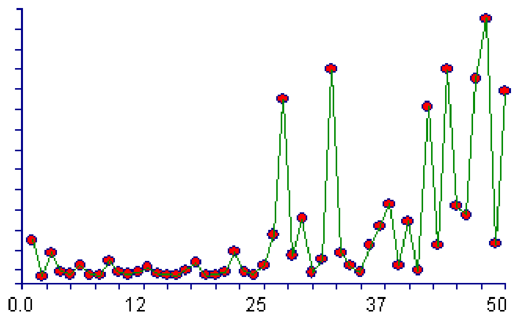


Fig. 5. χ^2 Control chart for χ_1 and χ_2 for presented part

REFERENCE

[1] Montgomery DC, 2001. Introduction to statistical quality control (4th ed.). New York: Wiley.

[2] Pugh GA, 1989. Synthetic neural networks for process control. Computers and Industrial Engineering. 17(1–4):24-26.

[3] Pugh, GA, 1991. A comparison of neural networks to SPC charts. Computers and Industrial Engineering, 21(1–4), 253-255.

[4] Hwarng, HB, 1995. Proper and effective training of a pattern recognizer for cyclic data. IIE Transactions. 27(6):746-756.

[5] Hwarng HB, Chong CW, 1995. Detecting process non-randomness through a fast and cumulative learning ART-based pattern recognizer. International Journal of Production Research 33(7):1817-1833.

[6] Pham DT, Chan AB, 1998. Control chart pattern recognition using a new type of self-organizing neural network. Proceedings of the Institution of Mechanical Engineers Part I, 12, I03597.

[7] Pham DT, Chan AB, 2001. Unsupervised adaptive resonance theory neural networks for control chart pattern recognition. Proceedings of the Institution of Mechanical Engineers. Part B, 215, B04999.

[8] Sagiroglu S, Besdok E, Erler M, 2000. Control chart pattern recognition using artificial neural networks. Turkish Journal of Electrical Engineering and Computer Sciences. 8(2) [c TUBITAK].

[9] Chiu CC, Chen MK, Lee KM, 2001. Shifts recognition in correlated process data using a neural network. International Journal of Systems Science. 32(2):137-143.

[10] Niaki, STA, Abbasi B, 2005. Fault diagnosis in multivariate control charts using artificial neural networks. International Quality and Reliability Engineering. 21:825-840.

[11] Guh R-S, 2005. A hybrid learning-based model for on-line detection and analysis of control chart patterns. Computers & Industrial Engineering. 49:35-62.

[12] Chen Z, Lu S, Lam S, 2007. A hybrid system for SPC concurrent pattern recognition. Advanced Engineering Informatics. 21:303-310.

[13] Guh, R-S, Shiue Y-R, 2008. An effective application of decision tree learning for on-line detection of mean shifts in multivariate control charts. Computers & Industrial Engineering. 55:475-493.

[14] Chen L-H, Wang T-Y, 2004. Artificial neural networks to classify mean shifts from multivariate χ^2 chart signals. Computers & Industrial Engineering, 47:195-205.

[15] Yu J-b, Xi L-f, 2009. A neural network ensemble-based model for on-line monitoring and diagnosis of out-of-control signals in multivariate manufacturing processes. Expert Systems with Applications. 36:909-921.

[16] El-Midany TT, El-Baz MA, Abd-Elwahed MS, 2010. A proposed framework for control chart pattern recognition in multivariate process using artificial neural networks, Expert Systems with Applications. 37:035-1042.

[17] Battiti R, 1994. Using mutual information for selecting features in supervised neural net learning. IEEE Transaction on Neural Networks. 5:537-550.

[18] Smith AE, 1994. X-bar and R control chart interpretation using neural computing. International Journal of Production Research. 32:309–320.

[19] Hassan A, Shariff Nabi Baksh M, Shaharoun, AM, Jamaluddin H, 2003. Improved SPC chart pattern recognition using statistical features. International Journal of Production Research. 41(7):1587-1603.

[20] Zurada JM, (1992). Introduction to artificial neural systems. Boston: PWS.

[21] Zorriassatine F, Tannock JDT, Brien CO, 2003. Using novelty detection to identify abnormalities caused by mean shifts in bivariate processes. Computers & Industrial Engineering. 44:385-408.

[22] Guh RS, 2004. Optimizing feedforward neural networks for control chart pattern recognition through genetic algorithms. International Journal of Pattern Recognition and Artificial Intelligence. 18(2):75-99.

[23] Barghash MA, Santarisi NS, 2004. Pattern recognition of control charts using artificial neural networks–analyzing the effect of the training parameters. Journal of Intelligent Manufacturing. 15:35-644.

[24] Principe JC, Euliano NR, Lefebvre WC, 2000. Neural and adaptive systems: Fundamentals through simulations. New York: John Wiley.

[25] Rumelhart DE, McClelland JL, PDP Research Group, 1986. Parallel distributed processing: Explorations in the microstructure of cognition. MA. Cambridge: MIT Press.

[26] Cheng CS, 1995. A multi-layer neural network model for detecting changes in the process mean. Computers and Industrial Engineering. 28:51-61.

[27] Kecman V, 2001. Learning and soft computing: Support vector machines, neural networks, and fuzzy logic models. Cambridge, MA: MIT press.